REMARKS/ARGUMENTS

Claims 1-38 are pending in this application, of which claims 1, 3, 4, 6, 11, 22, 24, and 27 are amended, and claims 5, 23, and 25 are cancelled. Claims 3-12 and 15 stand rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by U.S. Patent No. 5,481,722 ("Skinner"). Claims 22-26 stand rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by U.S. Patent No. 5,903,897 ("Carrier"). Claims 27, 29 and 31-38 stand rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by U.S. Patent No. 5,649,200 ("Leblang"). Claims 1-2 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over U.S. Patent No. 6,223,343 ("Hopwood") in view of Skinner. Claims 16-21 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Skinner in view of Hopwood. Claims 13-14 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Skinner in view of U.S. Patent No. 5,481,722 ("Howard"). Claims 28 and 30 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Leblang. Applicant respectfully requests reconsideration of the present application in light of the above recited amendments and below recited remarks.

The Claimed Invention

The present invention discloses systems and methods for managing code changes for software development. More specifically it is disclosed that:

> "[a]n aspect of the invention provides better ergonomics and increased speed in merging versioned documents by indicating differences between them at multiple different subdivisions or levels, such as line and character levels.
>
> Another aspect of the invention increases the reliability of merges between documents by comparing them not only for incompatible changes with respect to each other, but also by detecting possible alternative histories from a common parent document, and flagging these as potential conflicts.
>
> Another aspect increases flexibility in reverting to previous versions by removing changes made during an earlier version to be backed out from a current version while retaining changes made in a version later than the removed version but earlier than the current version.
>
> A further aspect of the invention increases the ability to integrate all the material pertaining to a change, by keeping it together in one place. This aspect associates both versioned and nonversioned documents for the same version together, so that they can be manipulated as a single unit.

> Yet a further aspect permits developers to work in together in constructing a new build of the software even while another build is being tested and patched. Multiple copies of the documents are made in different areas. While one set of copies is built and tested, private copies for each builder have previous changes removed, so that they can modify clean copies of the documents (Application, Summary of the Invention)."

Rejections Under 35 U.S.C. § 102(b)

1. Claims 3-12 and 15 stand rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by Skinner. Applicants respectfully traverse.

Skinner discloses a method and apparatus for merging change control delta structure files of a source module from a parent and a child development environment. Change deltas are created and propagated among the environments without any loss in change history (Skinner, Summary of the Invention).

With respect to ***newly amended*** independent claim 3, Skinner does not teach or suggest features recited in the claim, namely:

> "defining ***subdivisions within lines*** in both of the documents;
> for a current subdivision, comparing a current subdivision of one of the documents to a current subdivision of the other document and indicating any differences therebetween;
> repeating the . . . comparing step for further subdivisions of the documents within the current line . . .
> producing an output document indicating the differences found in the . . .comparing step."

Similarly, independent claim 8 recites similar comparing limitations as independent claim 3 directed to three types of nested subdivisions.

In the Office Action, the Examiner asserts that, in Skinner, "the comparing of line by line text is ***implicitly*** disclosed (Office Action, Page 3)." Thus, the Examiner concedes that line by line text comparison is not explicitly disclosed by Skinner. Furthermore, the Examiner asserts that, in Skinner, "the comparing of character by character type in a text line is ***implicitly*** disclosed (Office Action, Page 4)." Applicants disagree with both of these assertions. However, for the sake of argument, even if line by line comparison were somehow construed to be implicitly disclosed by Skinner, Applicants fail to see how character by character comparison could possibly, even implicitly, be disclosed by Skinner. Such alleged implicit disclosure would require two levels of abstraction from what is allegedly explicitly disclosed by Skinner. Furthermore, Applicants have reviewed Skinner in detail and are entirely unable to find any mention or implication of character by character comparison of a line. Applicants respectfully request that the Examiner cite where in Skinner there is any teaching or implication of character by character comparison.

With respect to claim 3, Skinner does not teach or suggest character by character comparison or any comparison of subdivisions of a line, as required by the claim.

With respect to claim 8, the Examiner alleges that characters are the third subdivision level required by the claim (sections allegedly being the first subdivision level, and lines allegedly being the second sub-division level). Skinner does not teach or suggest character by character comparison or the comparison of any three types of subdivisions, as required by the claim.

With respect to independent claim 11, Skinner does not teach or suggest features recited in the ***newly amended*** claim, namely:

> comparing both child documents with ***an identical version of*** the common parent document and indicating any possible conflicts between the child documents for portions of the child documents that are the same as each other; and
>
> producing a merged output document indicating both the actual and the possible conflicts.

In fact, Skinner teaches away from comparing children to ***an identical version of*** a common parent. Skinner discloses that two children may store a first revision of a parent. When a first child is reconciled with the parent, the modifications made in the first child are compared with the first revision of the parent, and the modifications to the first revision are incorporated into a second revision of the parent. When a second child attempts to reconcile with the parent, the modifications made in the second child are resynchronized and compared with the second revision of the parent. Thus, Skinner discloses that the first child is compared to the first revision of the parent while the second child is compared to the second revision of the parent (Skinner, Fig. 8; Col. 12, lines 29-51). The first revision of the parent and the second revision of the parent are clearly not ***an identical version of*** a "common parent".

Applicants respectfully submit that dependent claims 4, 6, 7, 9, 10, and 12 are patentable at least by reason of their dependency.

2. Claims 22-26 stand rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by Carrier. Applicants respectfully traverse.

Carrier discloses a software documentation release control system. A source module is attached to a corresponding release form primarily by engineers or developers. The release form is then assigned to a build. Documentation about the build such as, for example, a build log and a build report are stored in a project file directory (Carrier, Summary of the Invention).

With respect to ***newly amended*** independent claim 22, Carrier does not teach or suggest features recited in the claim, namely "storing the association file in a memory separately from the documents listed in the file." In fact, Carrier does not even mention storing an association file. With respect to former claim 25 (now incorporated into claim 22) the Examiner asserts that the list of approved release forms of Carrier reads on the association file. Even if this assertion is true, Carrier does not at all mention storing the list of approved release forms. The cited portion of Carrier (Col. 6, ln. 17-52) merely discloses that a list of open release forms may be displayed and that a list of release forms may be submitted to select release forms in connection with a build. Applicants request that the Examiner cite where in Carrier there is any mention of storing the list of approved release forms and, more particularly, storing the list of approved release forms in a memory separately from the documents listed in the file, as required by newly amended claim 22.

Applicants respectfully submit that dependent claims 24 and 26 are patentable at least by reason of their dependency.

3. Claims 27, 29 and 31-38 stand rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by Leblang. Applicants respectfully traverse.

Leblang discloses a dynamic rule-based version control system. Derived objects may be stored in a public VOB area or a private view corresponding to particular users. All objects in the private view are initially read only objects, but, if the users wish to edit the objects, then the objects may be "checked out". Upon completion of the editing, the objects may be "checked in". Each derived object may have a corresponding configuration record that contains an audit of the build process used to create the object (Leblang, Summary of the Invention).

The Examiner analogizes the configuration record of Leblang to the association file recited in independent claim 27. However, Leblang does not teach or suggest features of the association file recited in ***newly amended*** claim 27, namely:

> a plurality of entries each designating a version of one of the versioned documents ***to be incorporated in a change set***;
> at least one entry designating a nonversioned document pertaining to at least one of the versioned documents ***to be incorporated in the change set***.

Although the configuration record of Leblang ***audits*** elements previously used in a build, the configuration record does not ***designate*** elements to be incorporated in a change set, as required by claim 27. The configuration record is not generated until the build occurs and is merely a record of elements ***that have already been incorporated*** into the build (Leblang, Col. 25, lines 30-42).

With respect to independent claim 31, Leblang does not teach or suggest features of the claim, namely:

> "adding a set of build specific changes to files in the enlistment area;
> making local changes to a file; and
> thereafter, removing the build-specific changes from the enlistment files . . ."

In the Applicants' reply to the previous Office Action dated July 17, 2003, Applicants noted that Leblang does not mention adding and removing build specific changes. In the Examiner's Response to Arguments, the Examiner asserted that, "[s]ince the overall scheme by Leblang's method is the providing of a build based on a configuration file, the intermediate steps of synchronizing files would end up in making specific changes for the target build." Applicants fail to understand how simply stating that the overall scheme by Leblang's method is the providing of a build reads on the cited limitations of claim 31 above. Leblang arguably discloses that files which are checked out and checked in by a user may be approved to go into a build. However, even if a checked out file is approved for a build, then, at best, Leblang merely discloses that build specific changes are ***added*** to the checked out file. Absolutely nowhere in Leblang is there any teaching or suggestion of ***removing*** build specific changes prior to checking the file back in or at any time.

Applicants respectfully submit that dependent claims 29 and 32-38 are patentable at least by reason of their dependency. Accordingly, reconsideration and withdrawal of the 35 U.S.C. § 102(b) rejections are respectfully requested.

Rejections Under 35 U.S.C. § 103(a)

4. Claims 1-2 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Hopwood in view of Skinner. Applicants respectfully traverse.

Hopwood discloses a computer system and method to track and control element changes throughout application development. A revision management system includes an inventory group manager (IGM), a work group manager (WGM), and a version manager (VM). The IGM generates and manages one or more lists for each of one or more elements logically related to each other as one or more inventory groups. The IGM also maintains and manages the lists of the elements in the inventory group and provides new elements to be added to the inventory group. The WGM selects the elements relating to a project that require modification. and selects elements from the inventory groups to form a project work group. The VM tracks and manages the modifications to the elements in the project work group, and generates reports with respect to the modifications responsive to user selectable criteria (Hopwood, Summary of the Invention).

Applicants respectfully submit that the combination of Hopwood and Skinner do not teach or suggest features recited in ***newly amended*** claim 1, namely:

> "unmerging from a later version of one of the versioned documents a set of changes previous to a further set of changes ***while preserving the further set of changes***"

In the Examiner's Response to Applicants' Argument's, the Examiner asserts that reverting as disclosed by Hopwood reads on this limitation. However, reverting involves replacing a later version with a previous version. Thus, reverting involves removing a further set of changes to return a later version to its previous state (Hopwood, Col 28, lines 51-55). By contrast, unmerging as recited in claim 1 requires removing a set of changes previous to a existing further set of changes while preserving the further set of changes.

Applicants respectfully submit that dependent claim 2 is patentable at least by reason of its dependency.

5. Claims 16-21 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Skinner in view of Hopwood. Applicants respectfully traverse.

Skinner in view of Hopwood do not teach or suggest the selective unmerging feature recited in independent claims 16 and 20. The Examiner admits that Skinner does not disclose this feature (Office Action, page 10, paragraph 5). Furthermore, as discussed above with reference to claim 1, revert as disclosed by Hopwood is not analogous to this feature.

Applicants respectfully submit that dependent claims 17-19 and 21 are patentable at least by reason of their dependency.

6. Claims 13-14 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Skinner in view of U.S. Patent No. 5,481,722 ("Howard"). Applicants respectfully traverse. Applicants respectfully submit that claims 13 and 14 are dependent from independent claim 11 and are therefore patentable for the same reasons.

7. Claims 28 and 30 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Leblang. Applicants respectfully traverse. Applicants respectfully submit that claims 28 and 30 are dependent from independent claim 27 and are therefore patentable for the same reasons. Accordingly, reconsideration and withdrawal of the 35 U.S.C. § 103(a) rejections are respectfully requested.

CONCLUSION

In view of the above remarks, Applicants respectfully submit that the present application is in condition for allowance. Reconsideration of the application and an early Notice of Allowance are respectfully requested.

Date: April 16, 2004

Kenneth R. Eiferman
Registration No. 51,647

Woodcock Washburn LLP
One Liberty Place - 46th Floor
Philadelphia PA 19103
Telephone: (215) 568-3100
Facsimile: (215) 568-3439